AD-A051 491 TEXAS UNIV AT AUSTIN CENTER FOR NUMERICAL ANALYSIS F/G 12/1
THE ACCELERATED SSOR METHOD FOR SOLVING LARGE LINEAR SYSTEMS.(U)
MAY 77 L J HAYES, D M YOUNG DAHC04-74-G-0198
UNCLASSIFIED ITR-CNA-123 ARO-12301.6-M NL

1 OF 1
AD
A051491

END
DATE
FILMED
4-78
DDC

ARO 12301.6-M

Preliminary rept.,

12

THE ACCELERATED SSOR METHOD FOR SOLVING LARGE LINEAR SYSTEMS. PRELIMINARY REPORT

Linda J. Hayes ~~and~~ David M. Young

May 1977

ITR-CNA-123

48p.

Abstract

The symmetric SOR method (SSOR method) for solving the linear system $Au = b$ is considered. The basic properties of the SSOR method are summarized, and a procedure is given for estimating the optimum relaxation factor ω and the corresponding spectral radius of the SSOR matrix $\mathcal{S}_\omega$. Two procedures for accelerating the convergence of the SSOR method are considered, one based on conjugate gradient acceleration and the second based on the use of Chebyshev acceleration. Two versions of conjugate gradient acceleration are considered--the nonadaptive and the adaptive.

*Work on this paper was supported in part by grant DAHC04-74-G-0198 from the Army Research Office at The University of Texas at Austin.

CENTER FOR NUMERICAL ANALYSIS
THE UNIVERSITY OF TEXAS AT AUSTIN

406 262

THE ACCELERATED SSOR METHOD FOR SOLVING LARGE LINEAR SYSTEMS: PRELIMINARY REPORT

by

Linda J. Hayes and David M. Young

1. Introduction. In this paper we consider the symmetric SOR method (SSOR method) for solving the linear system

$$(1.1) \qquad Au = b$$

where A is a given real $N \times N$ matrix which is symmetric and positive definite and where b is a given real $N \times 1$ column matrix. We are primarily interested in cases where the matrix A is large and sparse.

The basic properties of the SSOR method are summarized in Section 2. A procedure is given for estimating the optimum relaxation factor ω and the corresponding spectral radius of the SSOR matrix $\mathcal{S}_\omega$. These estimates are in terms of the spectral radius $S(B)$ of the Jacobi method and another quantity $\bar{\beta}$ which can usually be estimated to sufficient accuracy.

We consider two procedures for accelerating the convergence of the SSOR method. The first, which is described in Section 3, is based on conjugate gradient acceleration. The second, which is described in Section 4, is based on the use of Chebyshev acceleration. To carry out either scheme one must choose the relaxation factor ω. No additional parameters are required to apply conjugate gradient acceleration. We consider two versions of conjugate gradient acceleration: the nonadaptive version where the value of ω is fixed; and the adaptive version where one chooses a value of ω and improves it adaptively based on the results of the iterative process.

In order to apply Chebyshev acceleration it is necessary to assume values of ω and $S(\mathcal{B}_\omega)$. We consider three versions: the <u>nonadaptive</u> version where neither ω nor the estimate of $S(\mathcal{B}_\omega)$ varies; the <u>partially adaptive</u> version where ω is kept fixed but where improved values of $S(\mathcal{B}_\omega)$ are determined adaptively; and the <u>fully adaptive</u> version where improved values of both ω and $S(\mathcal{B}_\omega)$ are determined adaptively.

Preliminary tests have been carried out for a very simple model problem as described in Section 5. For these cases the adaptive procedures are very effective in the sense that the number of iterations required in each case was not substantially greater than was required using the optimum iteration parameters.

We expect that the accelerated SSOR method with adaptive parameter determination will prove effective in a much wider class of problems. Experiments described by Young [2] and by Benokraitis [5] indicate that the nonadaptive procedure is effective. Tests on similar cases based on the adaptive procedures are now being carried out.

<u>Acknowledgement</u>. The authors would like to acknowledge the contributions of Mr. Edward Schleicher in preliminary stages of the work. (See [12].)

Some of the numerical experiments involving the Chebyshev acceleration were carried out by James Sullivan in the Center for Numerical Analysis at UT Austin. We would also like to acknowledge the help of Roger Grimes (UT Austin) in reading the manuscript.

2. The basic SSOR method. The equation (1.1) can be written in the form

$$(2.1) \qquad u = Bu + c$$

where

$$(2.2) \qquad \begin{cases} B = I - D^{-1}A \\ c = D^{-1}b. \end{cases}$$

Here D is the diagonal matrix with the same diagonal elements as A. The matrix B corresponds to the Jacobi method of iteration.

The matrix A can be written in the form

$$(2.3) \qquad A = D - C_L - C_U$$

where C_L and C_U are strictly lower and strictly upper triangular matrices, respectively. Evidently

$$(2.4) \qquad B = L + U$$

where

$$(2.5) \qquad L = D^{-1}C_L \qquad U = D^{-1}C_U$$

and where L and U are strictly upper triangular and strictly lower triangular matrices, respectively.

The SSOR method can be defined in terms of the ordinary (forward) SOR method and the backward SOR method. The ordinary SOR method is defined by

$$u^{(n+1)} = \mathcal{L}_\omega u^{(n)} + k_\omega^{(F)} \tag{2.6}$$

where

$$\begin{cases} \mathcal{L}_\omega = (I - \omega L)^{-1}(\omega U + (1-\omega) I) \\ k_\omega^{(F)} = (I - \omega L)^{-1}\omega c. \end{cases} \tag{2.7}$$

The backward SOR method is defined by

$$u^{(n+1)} = \mathcal{U}_\omega u^{(n)} + k_\omega^{(B)} \tag{2.8}$$

where

$$\begin{aligned} \mathcal{U}_\omega &= (I - \omega U)^{-1}(\omega L + (1-\omega) I) \\ k_\omega^{(B)} &= (I - \omega U)^{-1}\omega c. \end{aligned} \tag{2.9}$$

One iteration of the SSOR method consists of one (forward) SOR iteration (to get $u^{(n+\frac{1}{2})}$) and one backward SOR iteration to get $u^{(n+1)}$. Thus we have

$$\begin{aligned} u^{(n+\frac{1}{2})} &= \mathcal{L}_\omega u^{(n)} + k_\omega^{(F)} \\ u^{(n+1)} &= \mathcal{U}_\omega u^{(n+\frac{1}{2})} + k_\omega^{(B)}. \end{aligned} \tag{2.10}$$

If we eliminate $u^{(n+\frac{1}{2})}$ we obtain

$$u^{(n+1)} = \mathcal{S}_\omega u^{(n)} + k_\omega \tag{2.11}$$

where

$$\begin{cases} \mathcal{S}_\omega = \mathcal{U}_\omega \mathcal{L}_\omega \\ k_\omega = \omega(2-\omega)(I-\omega U)^{-1}(I-\omega L)^{-1}c. \end{cases} \tag{2.12}$$

It is easy to show that

$$\mathcal{S}_\omega = I - \omega(2-\omega)\ (I-\omega U)^{-1}(I-\omega L)^{-1}D^{-1}A \tag{2.13}$$

$$= I - \frac{2-\omega}{\omega}\left(\frac{1}{\omega}D - C_U\right)^{-1}D\left(\frac{1}{\omega}D - C_L\right)^{-1}A$$

$$= I - Q^{-1}A$$

where

$$Q = \hat{c}W^TW \tag{2.14}$$

and where

$$\hat{c} = \frac{\omega}{2-\omega} \tag{2.15}$$

$$W = D^{-\frac{1}{2}}\left(\frac{1}{\omega}D - C_U\right). \tag{2.16}$$

Our analysis of the convergence properties of the SSOR method will be based on that given in Young [1974]. Let $m(B)$ and $M(B)$ be the smallest and largest eigenvalues of the Jacobi matrix B. We remark that $m(B) \leq 0$ and $M(B) \geq 0$. Furthermore, $M(B) < 1$ since A is symmetric and positive definite. In addition, the eigenvalues of the SSOR matrix, $\mathcal{S}_\omega$, are real, non-negative, and less than unity.

Let $\bar{\beta}$, $\bar{M}$, and $\underline{m}$ be numbers such that

$$\begin{cases} m(B) \geq \underline{m} \geq -2\sqrt{\bar{\beta}} \\ M(B) \leq \bar{M} \leq 2\sqrt{\bar{\beta}} \\ \bar{M} < 1 \\ S(LU) \leq \bar{\beta} \end{cases} \tag{2.17}$$

It can be shown that $S(B) \leq 2\sqrt{\bar{\beta}}$; hence if we have a bound $\bar{M}$ for $M(B)$ such that $\bar{M} > 2\sqrt{\bar{\beta}}$ we can replace $\bar{M}$ by $2\sqrt{\bar{\beta}}$. Similarly, if $\underline{m} < -2\sqrt{\bar{\beta}}$ we can replace $\underline{m}$ by $-2\sqrt{\bar{\beta}}$.

It can be shown that

$$S(\mathcal{L}_\omega) \leq \begin{cases} 1 - \omega(2-\omega)\dfrac{1-\bar{M}}{1-\omega\bar{M}+\omega^2\bar{\beta}} & \text{if } \bar{\beta} \geq \frac{1}{4} \text{ or if } \bar{\beta} < \frac{1}{4} \text{ and } \omega \leq \omega^* \\[2ex] 1 - \omega(2-\omega)\dfrac{1-\underline{m}}{1-\omega\underline{m}+\omega^2\bar{\beta}} & \text{if } \bar{\beta} < \frac{1}{4} \text{ and } \omega > \omega^*. \end{cases} \tag{2.18}$$

Here for $\bar{\beta} < \frac{1}{4}$ we define ω^* by

$$\omega^* = \frac{2}{1+\sqrt{1-4\bar{\beta}}}. \tag{2.19}$$

Moreover, the bound (2.18) is minimized if we let

$$\omega_1 = \begin{cases} \dfrac{2}{1+\sqrt{1-2\bar{M}+4\bar{\beta}}} & \text{if } \bar{M} \leq 4\bar{\beta} \\[2ex] \dfrac{2}{1+\sqrt{1-4\bar{\beta}}} = \omega^* & \text{if } \bar{M} \geq 4\bar{\beta}. \end{cases} \tag{2.20}$$

The corresponding bound on $S(\mathcal{L}_{\omega_1})$ is given by

$$(2.21)\qquad S(\mathscr{B}_{\omega_1}) \le \begin{cases} \dfrac{1 - \dfrac{1-\bar{M}}{\sqrt{1-2\bar{M}+4\bar{\beta}}}}{1 + \dfrac{1-\bar{M}}{\sqrt{1-2\bar{M}+4\bar{\beta}}}}, & \text{if } \bar{M} \le 4\bar{\beta} \\[2ex] \dfrac{1-\sqrt{1-4\bar{\beta}}}{1+\sqrt{1-4\bar{\beta}}} = \omega^* - 1, & \text{if } \bar{M} \ge 4\bar{\beta}, \end{cases}$$

and the eigenvalues λ of $\mathscr{B}_{\omega_1}$ will be in the range $0 \le \lambda \le S(\mathscr{B}_{\omega_1})$.

We shall refer to the value of ω_1 given by (2.20) as a "good" value of ω. Of course, ω_1 is not necessarily the true optimum value in the sense of minimizing $S(\mathscr{B}_\omega)$.

For example, if the matrix A is derived from a 5-point difference equation corresponding to a self-adjoint elliptic partial differential equation

$$(2.22)\qquad L[u] = (Au_x)_x + (Cu_y)_y + Fu = G,$$

and if we rewrite the difference equation in the form which corresponds to the Jacobi method

$$(2.23)\qquad \begin{aligned} u(x,y) &= \beta_1(x,y)u(x+h,y) + \beta_2(x,y)u(x,y+h) + \beta_3(x,y)u(x-h,y) \\ &\quad + \beta_4(x,y)u(x,y-h) + \tau(x,y) \end{aligned}$$

where

$$(2.24)\qquad \begin{cases} \beta_1(x,y) = \dfrac{A(x+\frac{h}{2},y)}{S(x,y)}, & \beta_2(x,y) = \dfrac{C(x,y+\frac{h}{2})}{S(x,y)} \\[2ex] \beta_3(x,y) = \dfrac{A(x-\frac{h}{2},y)}{S(x,y)}, & \beta_4(x,y) = \dfrac{C(x,y-\frac{h}{2})}{S(x,y)} \end{cases}$$

and

$$(2.25)\quad \begin{cases} S(x,y) = A(x+\frac{h}{2},y) + A(x-\frac{h}{2},y) + C(x,y+\frac{h}{2}) + C(x,y-\frac{h}{2}) - h^2F(x,y) \\ \tau(x,y) = -h^2\,\frac{G(x,y)}{S(x,y)}\,, \end{cases}$$

then the formulas for the SSOR method are given by

$$(2.26)\quad \begin{cases} u^{(n+\frac{1}{2})}(x,y) = \omega\{\beta_1 u^{(n)}(x+h,y) + \beta_2 u^{(n)}(x,y+h) + \beta_3 u^{(n+\frac{1}{2})}(x-h,y) \\ \qquad + \beta_4 u^{(n+\frac{1}{2})}(x,y-h) + \tau(x,y)\} + (1-\omega)u^{(n)}(x,y) \\ u^{(n+1)}(x,y) = \omega\{\beta_1 u^{(n+1)}(x+h,y) + \beta_2 u^{(n+1)}(x,y+h) + \beta_3 u^{(n+\frac{1}{2})}(x-h,y) \\ \qquad + \beta_4 u^{(n+\frac{1}{2})}(x,y-h) + \tau(x,y)\} + (1-\omega)u^{(n+\frac{1}{2})}(x,y)\,. \end{cases}$$

If one uses the natural ordering[‡], $\bar{\beta}$ can be calculated as

$$(2.27)\quad \bar{\beta} = \max\{\beta_3(x,y)\,[\beta_1(x-h,y) + \beta_2(x-h,y)\,] + \beta_4(x,y)\,[\beta_1(x,y-h) + \beta_2(x,y-h)\,]\}.$$

An upper bound $\bar{M}$ for $M(B)$ can be obtained from (6.58), page 1037 of Young and Gregory [1973].[†] The relaxation parameter, ω, is calculated using (2.20).

Application to the Model Problem P

We now consider the model problem defined by

$$(2.28)\qquad u_{xx} + u_{yy} = -1$$

in the unit square $0 \leq x \leq 1$, $0 \leq y \leq 1$ with $u = 0$ on the boundary. We will refer to this as "Model Problem P." We use a mesh size of h where h^{-1} is an integer.

[‡]The "natural ordering" for points in a region R_h is given by (x',y') follows (x,y) if $y' > y$ or else $y' = y$ and $x' > x$. Thus the points of R_h are treated row by row from left to right starting at the bottom row.

[†]This bound may not be good enough in some cases. In that case an adaptive procedure may be used.

From (2.27) we get

(2.29) $$\bar{\beta} = \frac{1}{4} .$$

From (6.58), page 1037, of [1] we get

(2.30) $$M(B) = \cos \pi h.$$

From (2.20) we have

(2.31) $$\omega_1 = \frac{2}{1 + \sqrt{2(1-M(B))}}$$

which is very close to the optimum ω , namely ω_b , for the SOR method. Moreover, by (2.21) we have

(2.32) $$S(\mathcal{S}_{\omega_1}) \le \frac{1 - \sqrt{\frac{1-M(B)}{2}}}{1 + \sqrt{\frac{1-M(B)}{2}}} .$$

For Model Problem P, it can be shown (see, for instance, Young [1974]) that

(2.33) $$S(LU) \le \frac{1}{4} \cos^2 \frac{\pi h}{2} .$$

Using $\frac{1}{4} \cos^2 \frac{\pi h}{2}$ for $\bar{\beta}$ we get

(2.34) $$\omega_1 = \frac{2}{1 + \sqrt{3} \sin \frac{\pi h}{2}}$$

and

(2.35) $$S(\mathcal{S}_{\omega_1}) \le \frac{1 - \frac{2}{\sqrt{3}} \sin \frac{\pi h}{2}}{1 + \frac{2}{\sqrt{3}} \sin \frac{\pi h}{2}}$$

The SSOR method requires about twice as much work per iteration as the SOR method and is about half as fast as the SOR method. This, at first sight, would seem to preclude the use of SSOR. However, since all of the eigenvalues of the matrix $\mathcal{S}_\omega$ are real, nonnegative, and less than unity, Young [1974], it is possible to accelerate the basic SSOR method using semi-iteration or variable extrapolation as shown below. The resulting method is faster by an order-of-magnitude than the SOR method. In order for this gain to be possible it is sufficient that $S(LU) - \frac{1}{4}$ be of the same order-of-magnitude, in some sense, as $1 - M(B)$.

3. Conjugate Gradient Acceleration

Let us now apply conjugate gradient acceleration to the SSOR method defined by (2.11). We note that $W\mathcal{S}_\omega W^{-1}$ is a symmetric matrix, where W is given by (2.16). Moreover, as stated in Section 2, the eigenvalues of $\mathcal{S}_\omega$ are real, non-negative, and less than unity.

We define the SSOR-CG method, i.e., the conjugate gradient acceleration procedure based on the SSOR method, by*

$$u^{(n+1)} = \rho_{n+1}\{\gamma_{n+1}\delta^{(n)} + u^{(n)}\} + (1 - \rho_{n+1})u^{(n-1)} \tag{3.1}$$

where $\delta^{(n)}$ is the pseudo-residual vector defined by

$$\delta^{(n)} = \mathcal{S}_\omega u^{(n)} + k_\omega - u^{(n)}, \tag{3.2}$$

and the γ's are given by

$$\gamma_{n+1} = (1 - RQ_{n+1})^{-1} \tag{3.3}$$

$$RQ_{n+1} = \frac{(W\delta^{(n)}, W\mathcal{S}_\omega \delta^{(n)})}{(W\delta^{(n)}, W\delta^{(n)})}. \tag{3.4}$$

Thus RQ_{n+1} is the Rayleigh Quotient of the vector $W\delta^{(n)}$ and the matrix WGW^{-1}. The $\{\rho_n\}$ are given by

$$\begin{cases} \rho_1 = 1 \\ \rho_{n+1} = \left[1 - \dfrac{\gamma_{n+1}}{\gamma_n} \dfrac{(W\delta^{(n)}, W\delta^{(n)})}{(W\delta^{(n-1)}, W\delta^{(n-1)})} \dfrac{1}{\rho_n}\right]^{-1}. \end{cases} \tag{3.5}$$

---

*The formulas can be derived from the analysis of Concus, Golub, and O'Leary [6]. See also Young [11].

The conjugate gradient acceleration procedure has the remarkable property that

$$(3.7) \qquad \|[\epsilon^{(n)}]_{CG}\|_{A^{\frac{1}{2}}} \leq \|[\epsilon^{(n)}]_{SI}\|_{A^{\frac{1}{2}}}$$

where $[\epsilon^{(n)}]_{CG} = u^{(n)} - \bar{u}$ ($\bar{u}$ is the exact solution of (1.1)) corresponds to conjugate gradient acceleration, and $[\epsilon^{(n)}]_{SI}$ is the error vector corresponding to <u>any</u> semi-iterative method based on the SSOR method. Note that for any vector v we have

$$(3.8) \qquad \|v\|_{A^{\frac{1}{2}}} = \sqrt{(A^{\frac{1}{2}}v, A^{\frac{1}{2}}v)} = \sqrt{(v, Av)} \ .$$

Thus $\|v\|_{A^{\frac{1}{2}}}$ could be computed without too much difficulty.

Since $W(I-\mathcal{S}_\omega)W^{-1}$ (where W is given by (2.16)) and $A^{\frac{1}{2}}(I-\mathcal{S}_\omega)A^{-\frac{1}{2}}$ are symmetric and positive definite, we could use either W or $A^{\frac{1}{2}}$ for W in (3.4) and (3.5). The choice $W = D^{-\frac{1}{2}}(\frac{1}{\omega}D - C_U)$ minimizes the $A^{\frac{1}{2}}$-norm of the error, and the choice $W = A^{\frac{1}{2}}$ minimizes some other norm of the error at each step. An inefficient but attractive scheme would be to use $W = D^{\frac{1}{2}}(\frac{1}{\omega}D - C_U)$ in the iterative and adaptive procedures and then use the $A^{\frac{1}{2}}$-norm for the stopping procedure. One would then be using the norm which is being minimized in the stopping tests. This would involve an extra matrix/vector operation.

Recursive Pseudo-residuals

We now describe an alternative formulation of the SSOR-CG method defined by (3.1)-(3.5). This alternative formulation was brought to our attention by Alan Cline of The University of Texas at Austin. It has been used by Dr. Gene Golub of Stanford University and others.

In the formulation (3.1)-(3.5) one is required to do two matrix/vector multiplications involving $\mathcal{S}_\omega$ each iteration, namely, $\mathcal{S}_\omega u^{(n)}$ in (3.2) and $\mathcal{S}_\omega \delta^{(n)}$ in (3.4). We can reduce this to one such matrix/vector multiplication by computing the $\delta^{(n)}$ recursively. Thus by (3.1) and (3.2) we have

$$\delta^{(n+1)} = \rho_{n+1}[\gamma_{n+1}\mathcal{S}_\omega \delta^{(n)} + (1-\gamma_{n+1})\delta^{(n)}] + (1-\rho_{n+1})\delta^{(n-1)} . \tag{3.8'}$$

The calculations could be done as follows. Given $\mathcal{S}_\omega \delta^{(n)}$, $\delta^{(n)}$, $\delta^{(n-1)}$, $u^{(n)}$, $u^{(n-1)}$, and ρ_n, we could compute the following:

γ_{n+1} by (3.3)

ρ_{n+1} by (3.5)

$u^{(n+1)}$ by (3.1)

$\delta^{(n+1)}$ by (3.8').

It should be noted, however, that this procedure involves several matrix/vector multiplications involving the matrix W. These can be eliminated as follows. Let $\Delta^{(n)}$ be the pseudo-residual associated with the SOR method. Thus $\Delta^{(n)}$ is given by

$$\Delta^{(n)} = \mathcal{L}_\omega u^{(n)} + k_\omega^{(F)} - u^{(n)} \tag{3.9}$$

and $\delta^{(n)}$ is given by

$$\delta^{(n)} = \mathcal{U}_\omega\{\Delta^{(n)} + u^{(n)}\} + k_\omega^{(B)} - u^{(n)} . \tag{3.9'}$$

It can be shown* that

(3.10) $$W(I-\mathcal{B}_\omega) = \frac{1}{\hat{c}} D^{\frac{1}{2}}(I-\mathcal{L}_\omega)$$

(3.11) $$W\delta^{(n)} = \frac{1}{\hat{c}} D^{\frac{1}{2}}\Delta^{(n)} .$$

Thus we have

(3.12) $$(W\delta^{(n)}, W\delta^{(n)}) = \frac{1}{\hat{c}^2}(D^{\frac{1}{2}}\Delta^{(n)}, D^{\frac{1}{2}}\Delta^{(n)}) .$$

Moreover, the $\Delta^{(n)}$ and the $\delta^{(n)}$ can be determined recursively by

(3.13) $$\Delta^{(n+1)} = \rho_{n+1}\left[\gamma_{n+1}(\mathcal{L}_\omega - I)\delta^{(n)} + \Delta^{(n)}\right] + (1-\rho_{n+1})\Delta^{(n-1)}$$

(3.14) $$\delta^{(n+1)} = \rho_{n+1}\left[\gamma_{n+1}\mathcal{U}_\omega\mathcal{L}_\omega\delta^{(n)} + (1-\gamma_{n+1})\delta^{(n)}\right] + (1-\rho_{n+1})\delta^{(n-1)} .$$

The ρ_{n+1} and γ_{n+1} are given by

(3.15) $$\gamma_{n+1} = \frac{(D^{\frac{1}{2}}\Delta^{(n)}, D^{\frac{1}{2}}\Delta^{(n)})}{(D^{\frac{1}{2}}\Delta^{(n)}, D^{\frac{1}{2}}(I-\mathcal{L}_\omega)\delta^{(n)})}$$

(3.16) $$\rho_{n+1} = \begin{cases} 1 & , \text{ if } n = 0 \\ \left[1 - \dfrac{\gamma_{n+1}}{\gamma_n} \dfrac{(D^{\frac{1}{2}}\Delta^{(n)}, D^{\frac{1}{2}}\Delta^{(n)})}{(D^{\frac{1}{2}}\Delta^{(n-1)}, D^{\frac{1}{2}}\Delta^{(n-1)})} \dfrac{1}{\rho_n}\right]^{-1} & , \text{ if } n \geq 1. \end{cases}$$

A procedure based on these formulas is described in the flow chart of Figure 3.1. We will defer discussion of the stopping tests until later.

---

*See Hageman and Young [1977], Chapter D.

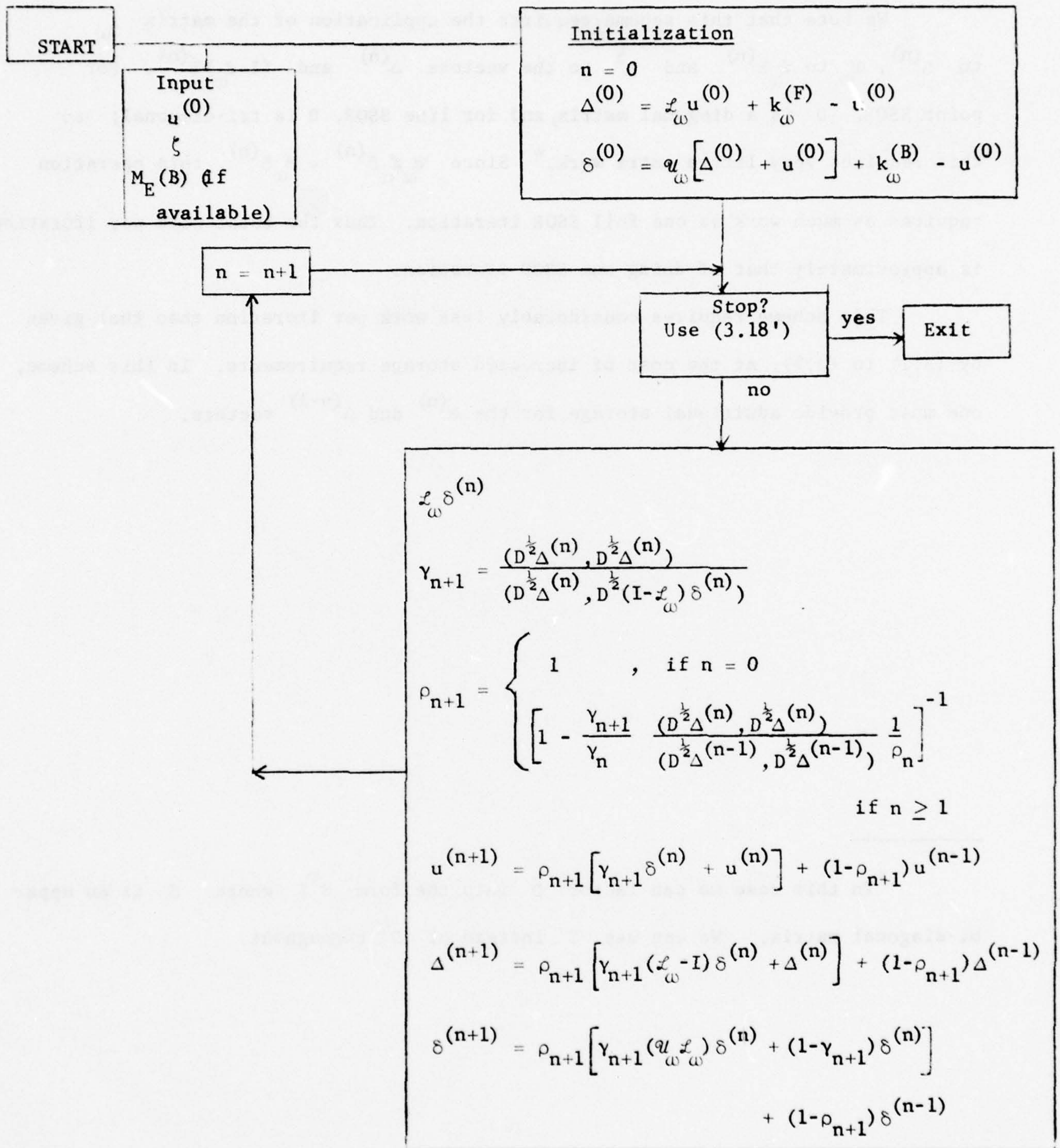

Figure 3.1. The SSOR-CG Method with Recursive Pseudo-residuals.

We note that this scheme requires the application of the matrix $\mathcal{L}_\omega$ to $\delta^{(n)}$, $\mathcal{U}_\omega$ to $\mathcal{L}_\omega \delta^{(n)}$ and $D^{\frac{1}{2}}$ to the vectors $\Delta^{(n)}$ and $(I-\mathcal{L}_\omega)\delta^{(n)}$. For point SSOR, D is a diagonal matrix, and for line SSOR, D is tri-diagonal; so this requires very little extra work.* Since $\mathcal{U}_\omega \mathcal{L}_\omega \delta^{(n)} = \mathcal{S}_\omega \delta^{(n)}$, this operation requires as much work as one full SSOR iteration. Thus the total work per iteration is approximately that of doing one SSOR iteration.

This scheme requires considerably less work per iteration than that given by (3.1) to (3.5), at the cost of increased storage requirements. In this scheme, one must provide additional storage for the $\Delta^{(n)}$ and $\Delta^{(n-1)}$ vectors.

---

*In this case we can factor D into the form S^TS where S is an upper bi-diagonal matrix. We can use S instead of $D^{\frac{1}{2}}$ throughout.

Stopping Tests

Ideally, we would like to require that our final approximation $u^{(n)}$ to the exact solution $\bar{u}$ of (1.1) satisfies

$$\boxed{\frac{\|u^{(n)} - \bar{u}\|_{D^{\frac{1}{2}}}}{\|\bar{u}\|_{D^{\frac{1}{2}}}} \leq \zeta} \tag{3.17}$$

It can be shown (see Hageman and Young [1977]) that (3.17) holds if

$$\boxed{\sqrt{\frac{\omega}{2-\omega}}\sqrt{\frac{1}{1-M(B)}}\;\frac{1}{1-S(\mathcal{S}_\omega)}\;\frac{\|\delta^{(n)}\|_W}{\|\bar{u}\|_{D^{\frac{1}{2}}}} \leq \zeta} \tag{3.18}$$

We will show below how we can get estimates $S_E(\mathcal{S}_\omega)$ of $S(\mathcal{S}_\omega)$ while carrying out the SSOR-CG method. If a good estimate $M_E(B)$ of $M(B)$ is available we can use the stopping test

$$\boxed{\sqrt{\frac{2-\omega}{\omega}}\sqrt{\frac{1}{1-M_E(B)}}\;\frac{1}{1-S_E(\mathcal{S}_\omega)}\;\frac{\|\Delta^{(n)}\|_{D^{\frac{1}{2}}}}{\|u^{(n)}\|_{D^{\frac{1}{2}}}} \leq \zeta} \tag{3.18'}$$

where $\Delta^{(n)}$ is the SOR pseudo-residual. This follows by (3.11) and (2.15).

In the adaptive SSOR-CG method which we describe later in this section we will automatically get estimates ot $M(B)$. If, however, we are not using the adaptive procedure and if we do not have a good estimate of $M(B)$, then we can be satisfied with requiring that our final approximatinn $u^{(n)}$ satisfy

$$\boxed{\frac{\|u^{(n)} - \bar{u}\|_W}{\|\bar{u}\|_W} \leq \zeta} \quad . \tag{3.19}$$

It can be shown that this condition is satisfied if

$$\boxed{\frac{1}{1-S(\mathcal{S}_\omega)} \frac{\|\delta^{(n)}\|_W}{\|\bar{u}\|_W} \leq \zeta} \quad . \tag{3.20}$$

Since our iterative process yields estimates $S_E(\mathcal{S}_\omega)$ for $S(\mathcal{S}_\omega)$ we can use the test

$$\boxed{\frac{2-\omega}{\omega} \frac{1}{1-S_E(\mathcal{S}_\omega)} \frac{\|\Delta^{(n)}\|_{D^{\frac{1}{2}}}}{\|u^{(n)}\|_W} \leq \zeta} \tag{3.20'}$$

$\bar{u}$ has been replaced by $u^{(n)}$ in (3.18') and (3.20'). In practice, $\bar{u}$ is not available, and as n increases $\|u^{(n)}\|_{D^{\frac{1}{2}}}$ and $\|u^{(n)}\|_W$ become a good estimate of $\|\bar{u}\|_{D^{\frac{1}{2}}}$ and $\|\bar{u}\|_W$.

We remark that if one uses either (3.18') or (3.20') it is necessary to compute $\|u^{(n)}\|_{D^{\frac{1}{2}}}$, or $\|u^{(n)}\|_W$, after each iteration. However, when one has reached the point where $\|u^{(n)} - \bar{u}\|_{D^{\frac{1}{2}}}$, or $\|u^{(n)} - \bar{u}\|_W$, is small, it is not necessary to change $\|u^{(n)}\|_{D^{\frac{1}{2}}}$, or $\|u^{(n)}\|_W$. In practice, $\|u^{(n)}\|_W$ and $\|u^{(n)}\|_{D^{\frac{1}{2}}}$ are not changed once the numerator of the stopping test is small.

For our estimate $S_E(\mathcal{S}_\omega)$ we use

$$S_E(\mathcal{S}_\omega) = M(T_n) \tag{3.20''}$$

where $M(T_n)$ is the largest eigenvalue of the matrix T_n. Here T_n is given by*

$$T_n = \begin{bmatrix} \frac{-(1-\gamma_1)}{\gamma_1} & \sqrt{\frac{-(1-\rho_2)}{\rho_1\gamma_2\rho_2}} & 0 & .. & 0 & 0 \\ \sqrt{\frac{-(1-\rho_1)}{\gamma_1\gamma_2\rho_2}} & \frac{-(1-\gamma_2)}{\gamma_2} & \sqrt{\frac{-(1-\rho_3)}{\gamma_2\rho_2\gamma_3\rho_3}} & .. & 0 & 0 \\ & \sqrt{\frac{-(1-\rho_3)}{\gamma_2\rho_2\gamma_3\rho_3}} & & & & \\ . & . & . & .. & . & . \\ . & . & . & .. & . & . \\ 0 & 0 & 0 & .. & \frac{-(1-\gamma_{n-1})}{\gamma_{n-1}} & \sqrt{\frac{-(1-\rho_n)}{\gamma_{n-1}\rho_{n-1}\gamma_n\rho_n}} \\ 0 & 0 & 0 & .. & \sqrt{\frac{-(1-\rho_n)}{\gamma_{n-1}\rho_{n-1}\gamma_n\rho_n}} & \frac{-(1-\gamma_n)}{\gamma_n} \end{bmatrix} \tag{3.21}$$

---

*The idea of using the eigenvalues of a tri-diagonal matrix derived from the conjugate graduate method to estimate the spectral radius of an iteration matrix is used by Concus, Golub, and O'Leary. See also Kaniel [1966], Paige [1971], and O'Leary [1975].

Adaptive Determination of ω for the SSOR-CG Method

So far, in our discussion of the SSOR-CG method we have assumed that ω is fixed. Actually, since $S(\mathcal{S}_\omega)$ is a very slowly varying function of ω, it is not essential that the exact optimum value of ω be used. It would not be unreasonable to simply guess at a value of ω, say 1.8, and use the non-adaptive process described above. The sensitivity of $S(\mathcal{S}_\omega)$ to ω is much less than is the sensitivity of $S(\mathcal{L}_\omega)$.

However, it is possible to improve ω adaptively using the matrices T_n described above. Instead of working with ω, we actually work with $M_E(B)$, which is an estimate for $M(B)$. We assume that we have an upper bound $\bar{\beta}$ for $S(LU)$. (Such a bound should not be much greater than 1/4 if the method is to be effective.) We choose an estimate $M_E(B)$ such that $M_E(B) \leq M(B)$--if nothing better is available let $M_E(B) = 0$. We then compute ω, based on $M_E(B)$ and $\bar{\beta}$, by (2.20) (with $\bar{M}$ replaced by $M_E(B)$). We also compute $S_E(\mathcal{S}_\omega)$ by (2.21).

Before beginning with the adaptive process we choose an "adaptive factor" F such that $0 < F < 1$. A typical value of F is 3/4. The effectiveness of the process is quite insensitive to F. If we use too large a va ue of F, say F very close to unity, we will change parameters very often, which is inefficient.

With too small a value of F we will not change parameters often enough and we will be working with a poor value of ω.

After each iteration we compute $S' = M(T_n)$, and we change $M_E(B)$ if

$$\frac{\chi_1}{\chi_2} < F \tag{3.23}$$

where

$$\chi_1 = -\log \left[\frac{\Phi(S_E(\mathcal{S}_\omega))}{\Phi\left(\frac{S_E(\mathcal{S}_\omega)}{S'}\right)} \right] \tag{3.24}$$

$$\chi_2 = -\log \Phi(S'). \tag{3.25}$$

Here we define $\Phi(x)$ for $x \in [0,1]$ by

$$\Phi(x) = \frac{1 - \sqrt{1-x}}{1 + \sqrt{1-x}} . \tag{3.26}$$

This procedure is based on the fact that χ_1 is the asymptotic average rate of convergence of the SSOR-SI method based on the use of $S_E(\mathcal{S}_\omega)$, if the true spectral radius $S(\mathcal{S}_\omega)$ is equal to S', and χ_2 is the asymptotic average rate of convergence of the SSOR-SI method based on the use of $S_E(\mathcal{S}_\omega) = S'$, again assuming that the true spectral radius is equal to S'. If the ratio of these convergence rates is at least F, then we consider that the convergence is fast enough and we do not change parameters. If the ratio is less than F, then we consider that the convergence is too slow and we change the parameters ω and $S_E(\mathcal{S}_\omega)$ in order to obtain faster convergence.

If $\bar{\beta} < 1/4$ and if at any stage of the adaptive process we have

$$\frac{-\log \Phi(\omega^*-1)}{-\log \Phi(S_E(\mathcal{S}_\omega))} \geq F, \tag{3.27}$$

then we can settle on ω^* for ω and we do not need to even consider changing ω

any more. If (3.27) is satisfied we know that by using $\omega = \omega^*$ our asymptotic average rate of convergence will be at least F times that which we could estimate by our methods for any other ω. (It should be noted, however, that the actual average rate of convergence for some ω might be greater than we can predict.)

Having decided to change parameters we compute $(M_E(B))_{new}$ by

$$(M_E(B))_{new} = \max(M_E(B),\ M_E'(B),\ M_E''(B)) \tag{3.28}$$

where

$$M_E'(B) = \frac{(1 - S')(1 + \omega^2\bar{\beta}) - \omega(2-\omega)}{\omega(\omega - (1+S'))} \tag{3.29}$$

and*

$$M_E''(B) = \begin{cases} \dfrac{\|B\delta^{(n)}\|_{D^{\frac{1}{2}}}}{\|\delta^{(n)}\|_{D^{\frac{1}{2}}}}, & \text{if } |(m(B))| \le M(B) \\[2ex] \dfrac{(D^{\frac{1}{2}}\delta^{(n)}, D^{\frac{1}{2}}B\delta^{(n)})}{(D^{\frac{1}{2}}\delta^{(n)}, D^{\frac{1}{2}}\delta^{(n)})}, & \text{otherwise.} \end{cases} \tag{3.30}$$

New values of ω and $S_E(\mathcal{L}_\omega)$ are obtained by (2.20) and (2.21), respectively, and with $\bar{M} = (M_E(B))_{new}$.

We remark that the value of $M_E'(B)$ is obtained by setting the first form of the right member of (2.18) equal to S' and solving for $\bar{M}$. We note that the first form should be used since we can show that if $\bar{\beta} < 1/4$, then each ω that we use will not be greater than ω^*.

---

*If A has Property A or is an L-matrix, then $|m(B)| \le M(B)$.

Let us define the integer s as follows. The current value of ω has been (or will be) used for the first time in the computation of $u^{(s+1)}$. The formulas for the adaptive SSOR-CG method are the same as in the non-adaptive case except that

$$(3.31) \qquad \rho_{n+1} = \begin{cases} 1 & , \text{ if } n = s \\ \left[1 - \dfrac{\gamma_{n+1}}{\gamma_n} \dfrac{(D^{\frac{1}{2}}\Delta^{(n)}, D^{\frac{1}{2}}\Delta^{(n)})}{(D^{\frac{1}{2}}\Delta^{(n-1)}, D^{\frac{1}{2}}\Delta^{(n-1)})} \dfrac{1}{\rho_n} \right]^{-1} & , \text{ if } n \geq s+1. \end{cases}$$

The overall procedure is illustrated in Figure 3.2. The input data are $u^{(0)}$, ζ, $M_E(B)$, F, and $\bar{\beta}$. We set n and s equal to zero and compute ω and $S_E(\mathcal{S}_\omega)$ as well as the initial pseudo-residual vectors $\Delta^{(0)}$ and $\delta^{(0)}$. Next, if $n \neq 0$ we find $M(T_{n,s})$ where $T_{n,s}$ is defined by (3.32).* (This is not done if $n = 0$.) We then apply the stopping test (3.18') with $S(\mathcal{S}_\omega)$ replaced by $S_E(\mathcal{S}_\omega)$. If convergence has not occurred, we consider whether to change $M_E(B)$.

The "switch" α is initially set equal to α_1. However, each time we go to α_1, we test whether ω^* would be satisfactory. This is the test (3.27). If at any time (3.27) is satisfied, then we set α to α_2, and from then on $M_E(B)$ and ω are not changed.

Assuming that $\alpha = \alpha_1$ we then use the test (3.27) to determine whether $M_E(B)$ should be changed. If $M_E(B)$ is to be changed, we carry out the change using (3.28). We then compute new values of ω, $S_E(\mathcal{S}_\omega)$, $\Delta^{(n)}$, and $\delta^{(n)}$. We then proceed to calculate $\mathcal{L}_\omega \delta^{(n)}$, $\mathcal{L}_{n+1}$, ρ_{n+1}, $u^{(n+1)}$, $\Delta^{(n+1)}$, and $\delta^{(n+1)}$. We are then ready for the next iteration.

The computation of $\mathcal{L}_\omega \delta^{(n)}$, γ_{n+1}, ρ_{n+1}, $u^{(n+1)}$, $\Delta^{(n+1)}$, and $\delta^{(n+1)}$ is the same as in the nonadaptive case except that ρ_{n+1} is given by (3.31).

---

*The matrix $T_{n,s}$ is used instead of T_n since when we change $M_E(B)$ (after the s-th iteration), we essentially start the conjugate gradient process over.

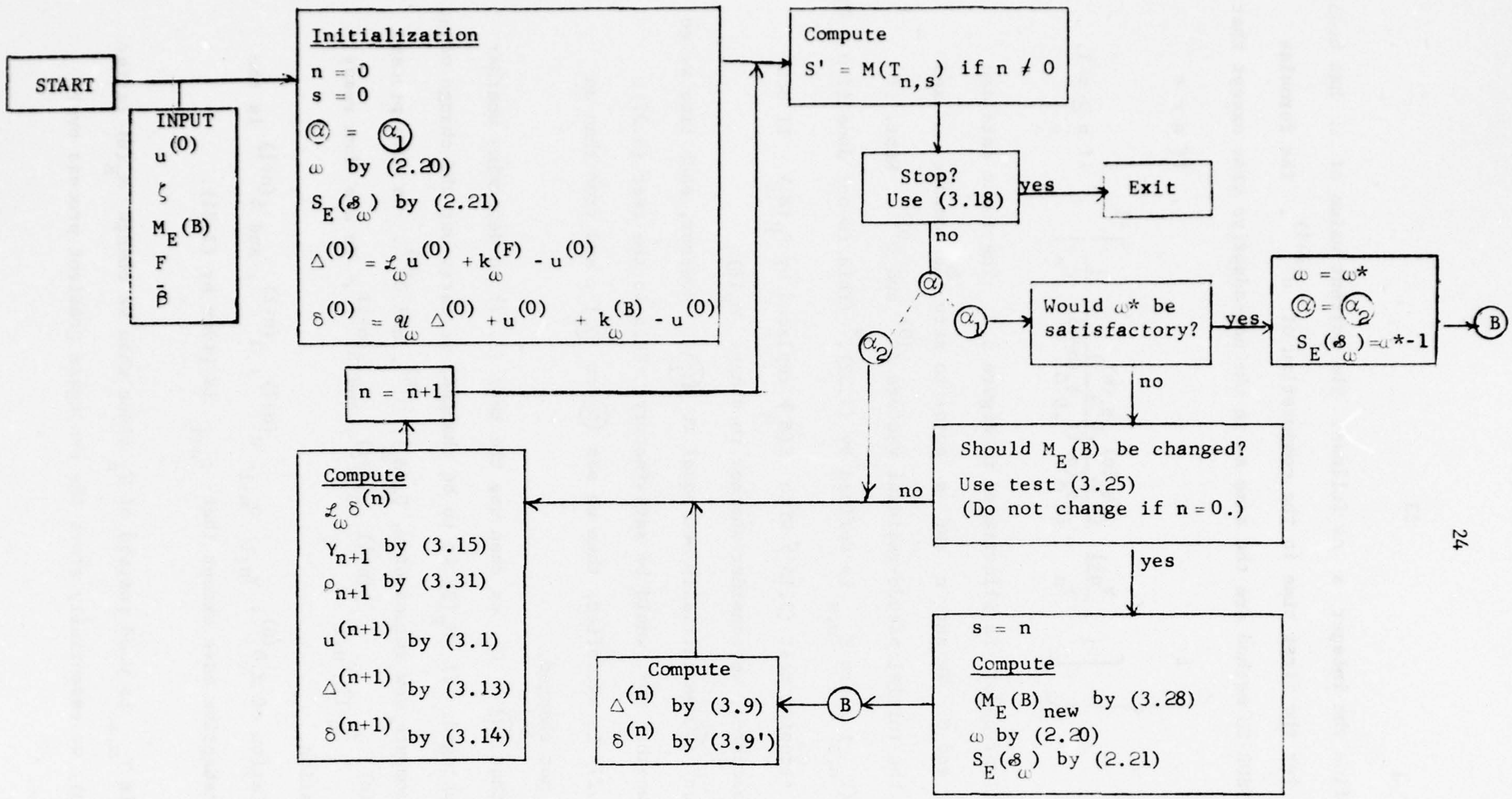

Figure 3.2. The Adaptive SSOR-CG Method with Recursive Pseudo-residuals

$$(3.32) \qquad T_{n,s} = \begin{bmatrix} \frac{-(1-\gamma_{s+1})}{\gamma_{s+1}} & \sqrt{\frac{-(1-\rho_{s+2})}{\gamma_{s+1}\gamma_{s+2}\rho_{s+2}}} & 0 & \cdot\cdot & 0 & 0 \\ \sqrt{\frac{-(1-\rho_{s+2})}{\gamma_{s+1}\gamma_{s+2}\rho_{s+2}}} & \frac{-(1-\gamma_{s+2})}{\gamma_{s+2}} & \sqrt{\frac{-(1-\rho_{s+3})}{\gamma_{s+2}\rho_{s+2}\gamma_{s+3}\rho_{s+3}}} & \cdot\cdot & 0 & 0 \\ \cdot & \sqrt{\frac{-(1-\rho_{s+3})}{\gamma_{s+2}\rho_{s+2}\gamma_{s+3}\rho_{s+3}}} & \cdot & \cdot\cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot\cdot & \cdot & \cdot \\ 0 & 0 & 0 & \cdot\cdot & \frac{-(1-\gamma_{n-1})}{\gamma_{n-1}} & \sqrt{\frac{-(1-\rho_n)}{\gamma_{n-1}\rho_{n-1}\gamma_n\rho_n}} \\ 0 & 0 & 0 & \cdot\cdot & \sqrt{\frac{-(1-\gamma_n)}{\gamma_{n-1}\rho_{n-1}\gamma_n\rho_n}} & \frac{-(1-\gamma_n)}{\gamma_n} \end{bmatrix}$$

4. Acceleration Based on Chebyshev Semi-iteration

As an alternative to the SSOR-CG method we can use Chebyshev semi-iteration to accelerate the SSOR method. The SSOR-SI method is defined by (see Young [2])

$$u^{(n+1)} = \rho_{n+1}\{\gamma_{n+1}\delta^{(n)} + u^{(n)}\} + (1-\rho_{n+1})u^{(n-1)} \tag{4.1}$$

where the pseudo-residual vector $\delta^{(n)}$ is given by (3.2) and where

$$\gamma_{n+1} = \frac{2}{2 - S(\mathcal{S}_\omega)} \tag{4.2}$$

and

$$\begin{cases} \rho_1 = 1 \\ \rho_2 = (1 - \frac{1}{2}\sigma^2)^{-1} \\ \rho_{n+1} = (1 - \frac{1}{4}\sigma^2\rho_n)^{-1}, \qquad n \geq 2. \end{cases} \tag{4.3}$$

Here

$$\sigma = \frac{S(\mathcal{S}_\omega)}{2-S(\mathcal{S}_\omega)}. \tag{4.4}$$

In order to apply the SSOR-SI method we need to choose ω and an estimate $S_E(\mathcal{S}_\omega)$ for $S(\mathcal{S}_\omega)$. Also, if we wish to use the stopping test (3.18') we need an estimate $M_E(B)$ of $M(B)$. Otherwise we use the stopping test (3.20').

In this section we will consider two adaptive procedures. The first, which we refer to as the "partially adaptive procedure," involves fixing ω and adaptively changing the estimate $S_E(\mathcal{S}_\omega)$ for $S(\mathcal{S}_\omega)$. The other procedure, which we refer to as the "fully adaptive procedure," involves changing ω and $S_E(\mathcal{S}_\omega)$ adaptively. The procedure where ω and $S_E(\mathcal{S}_\omega)$ are never changed is called the "non-adaptive procedure." For the fully adaptive procedure we work with $M_E(B)$ and hence can use the stopping test (3.18').

Partially Adaptive Procedure (ω fixed)

Let $S_E(\mathcal{B}_\omega)$ be the current estimate for $S(\mathcal{B}_\omega)$. Let us define the integer s as follows. The current value of $S_E(\mathcal{B}_\omega)$ has been (or will be) used starting with the $(s+1)^{st}$ iteration (to get $u^{(s+1)}$). The SSOR-SI method is defined by

(4.5) $$u^{(n+1)} = \rho_{n+1}\{\gamma_{n+1}\delta^{(n)} + u^{(n)}\} + (1 - \rho_{n+1})u^{(n-1)}$$

where the pseudo-residual vector $\delta^{(n)}$ is given by (3.2) and

(4.6) $$\gamma_{n+1} = \frac{2}{2 - S_E(\mathcal{B}_\omega)}$$

(4.7) $$\rho_{n+1} = \begin{cases} 1 \quad , & \text{if } n = s \\ (1 - \frac{1}{2}\sigma_E^2)^{-1}, & \text{if } n = s+1 \\ (1 - \frac{1}{4}\sigma_E^2\rho_n)^{-1}, & \text{if } n \geq s+2 \end{cases}$$

(4.8) $$\sigma_E = \frac{S_E(\mathcal{B}_\omega)}{2 - S_E(\mathcal{B}_\omega)}$$

We change $S_E(\mathcal{B}_\omega)$ whenever $n = 0$ or

(4.9) $$\frac{\|\delta^{(n)}\|_W}{\|\delta^{(s)}\|_W} \geq \left(\frac{2r^{p/2}}{1+r^p}\right)^F$$

where

(4.10) $$p = n - s,$$

and

$$(4.11)\qquad r = \frac{1-\sqrt{1-\sigma_E^2}}{1+\sqrt{1-\sigma_E^2}} = \left(\frac{1-\sqrt{1-S_E(\mathcal{S}_\omega)}}{1+\sqrt{1-S_E(\mathcal{S}_\omega)}}\right)^2$$

Here F is the adaptive factor discussed in Section 3 for the adaptive SSOR-CG procedure.

Having decided to change $S_E(\mathcal{S}_\omega)$ we let

$$(4.12)\qquad [S_E(\mathcal{S}_\omega))]_{new} = \max(S_E(\mathcal{S}_\omega),\ S'_E(\mathcal{S}_\omega),\ S''_E(\mathcal{S}_\omega))$$

where $S'_E(\mathcal{S}_\omega)$ and $S''_E(\mathcal{S}_\omega)$ are determined as follows. To get $S'_E(\mathcal{S}_\omega)$ we solve the Chebyshev equation

$$(4.13)\qquad \frac{\|\delta^{(n)}\|_W}{\|\delta^{(s)}\|_W} = \frac{2r^{p/2}}{1+r^p} \div \frac{2\hat{r}^{p/2}}{1+\hat{r}^p}$$

where r is given by (4.11) and

$$(4.14)\qquad \hat{r} = \left(\frac{1-\sqrt{1-(S_E(\mathcal{S}_\omega)/S'_E(\mathcal{S}_\omega))}}{1+\sqrt{1-(S_E(\mathcal{S}_\omega)/S'_E(\mathcal{S}_\omega))}}\right)^2 .$$

We compute $S''_E(\mathcal{S}_\omega)$ by the Rayleigh quotient

$$(4.15)\qquad S''_E(\mathcal{S}_\omega) = \frac{(W\delta^{(n)}, W\mathcal{S}_\omega\delta^{(n)})}{(W\delta^{(n)}, W\delta^{(n)})}$$

Having computed the new value of $S_E(\mathcal{S}_\omega)$ we let $s = n$ and then use (4.5)-(4.8) to continue the iteration process.

The overall computational procedure is illustrated in Figure 4.1. As in the case of the SSOR-CG method, we determine the pseudo-residual $\Delta^{(n)}$ for the SOR method for each iteration as well as the pseudo-residual $\delta^{(n)}$ for the SSOR method. This saves matrix/vector multiplications by W. At the beginning of each iteration we compute $\Delta^{(n)}$ and $\delta^{(n)}$. Usually this is done in a straightforward way, but if $n = s+1$ we use a special procedure as described below. We then carry out our stopping test. We use (3.18') if a good estimate $M_E(B)$ is available for $M(B)$. Otherwise we use (3.20').

If the process has not converged, we test whether $S_E(\mathcal{S}_\omega)$ should be changed. If the test (4.9) is satisfied or if $n = 0$, we compute $(S_E(\mathcal{S}_\omega))_{new} = \max(S_E(\mathcal{S}_\omega), S_E'(\mathcal{S}_\omega), S_E''(\mathcal{S}_\omega))$ where $S_E'(\mathcal{S}_\omega)$ is the solution of the Chebyshev equation and $S_E''(\mathcal{S}_\omega)$ is the Rayleigh quotient (4.15). With the new value of $S_E(\mathcal{S}_\omega)$ we compute ρ_{n+1}, γ_{n+1}, and $u^{(n+1)}$ and are ready for the next iteration.

We use a special procedure whereby one can avoid matrix/vector multiplications by W and, at the same time, can compute the Rayleigh quotient $S''_E(\mathcal{S}_\omega)$ each time $S_E(\mathcal{S}_\omega)$ is changed without an extra matrix/vector multiplication. When we decide to change $S_E(\mathcal{S}_\omega)$ we compute the vector

$$\tilde{u}^{(n+1)} = u^{(n)} + \delta^{(n)}. \tag{4.16}$$

It can be shown that

$$\tilde{\delta}^{(n+1)} = \mathcal{S}_\omega \tilde{u}^{(n+1)} + k_\omega - \tilde{u}^{(n+1)} = \mathcal{S}_\omega \delta^{(n)}. \tag{4.17}$$

Thus, for $S''_E(\mathcal{S}_\omega)$ we have by (3.11) (which holds for $\tilde{\delta}^{(n+1)}$ and $\tilde{\Delta}^{(n+1)}$)

$$S''_E(\mathcal{S}_\omega) = \frac{(W\delta^{(n)}, W\mathcal{S}_\omega \delta^{(n)})}{(W\delta^{(n)}, W\delta^{(n)})} = \frac{(D^{\frac{1}{2}}\Delta^{(n)}, D^{\frac{1}{2}}\tilde{\Delta}^{(n+1)})}{(D^{\frac{1}{2}}\Delta^{(n)}, D^{\frac{1}{2}}\Delta^{(n)})} \tag{4.18}$$

where

$$\tilde{\Delta}^{(n+1)} = \mathcal{L}_\omega \tilde{u}^{(n+1)} + k_\omega^{(F)} - \tilde{u}^{(n+1)}. \tag{4.19}$$

We note that to get the Rayleigh quotient $S''_E(\mathcal{S}_\omega)$ we have essentially had to do an extra SSOR iteration. However, we can recover this by using a special procedure to get $\Delta^{(n)}$ and $\delta^{(n)}$ on the next iteration. Thus it can be shown that if $n = s+1$ then

$$\begin{cases} \delta^{(n)} = \gamma_n \tilde{\delta}^{(n)} + (1-\gamma_n)\delta^{(n-1)} \\ \Delta^{(n)} = \gamma_n \tilde{\Delta}^{(n)} + (1-\gamma_n)\Delta^{(n-1)} \end{cases} \tag{4.20}$$

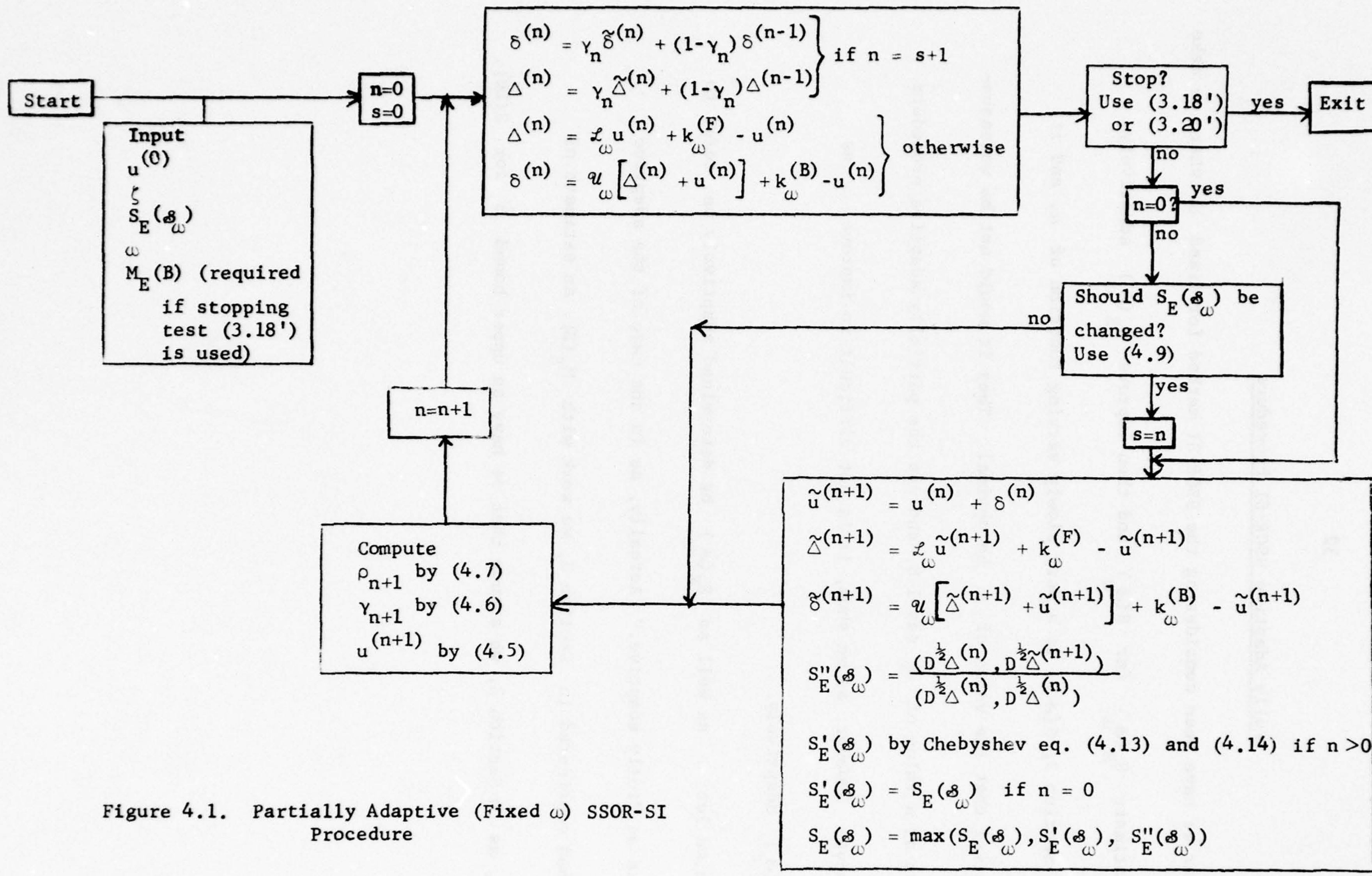

Figure 4.1. Partially Adaptive (Fixed ω) SSOR-SI Procedure

Fully Adaptive SSOR-SI Procedure

So far we have been considering the SSOR-SI method for fixed ω where we make an initial estimate $S_E(\mathcal{B}_\omega)$ for $S(\mathcal{B}_\omega)$ and then improve $S_E(\mathcal{B}_\omega)$ adaptively. As stated in Section 3, $S(\mathcal{B}_\omega)$ is a very slowly varying function of ω, and it is not essential that the value of ω be optimal. Thus it would not be unreasonable to guess at a value of ω, say 1.8, and use the partially adaptive procedure described above. However, as we show, it is not difficult to improve ω as well as $S_E(\mathcal{B}_\omega)$ adaptively.

When we let ω as well as $S_E(\mathcal{B}_\omega)$ be determined adaptively we refer to the procedure as "fully adaptive." Actually, as in the case of the adaptive SSOR-CG method considered in Section 3, we work with $M_E(B)$, an estimate of $M(B)$. Also, as in Section 3, we assume that we have an upper bound $\bar{\beta}$ for $S(LU)$.

The overall procedure for the fully adaptive SSOR-SI method is shown in Figure 4.2. To begin with, we estimate a value of $M_E(B)$ such that $0 \leq M_E(B) \leq M(B)$ and also such that $M_E(B) \leq 2\sqrt{\bar{\beta}}$. We compute ω by (2.20) with $\bar{M} = M_E(B)$ and $S_E(\mathcal{S}_\omega)$ by (2.21).

At the beginning of each iteration we compute $\Delta^{(n)}$ and $\delta^{(n)}$. We then apply the stopping test (3.18'). If convergence has not occurred, we proceed to switch α. Initially, $\alpha = \alpha_1$. However, if at any time ω^* can be shown to be satisfactory (using (3.27)) we let $\alpha = \alpha_2$ and from then on we will not consider changing $M_E(B)$. We then apply test (4.9) to see whether $M_E(B)$ should be changed. If so, or if $n = 0$, we proceed as follows. We compute a new value of $S_E(\mathcal{S}_\omega)$ as indicated. The value of $S_E''(\mathcal{S}_\omega)$ given is the Rayleigh quotient, as was shown in our discussion of the partially adaptive process. Having found a new value of $S_E(\mathcal{S}_\omega)$, we proceed to get $M_E(B)$ by (3.28). Then we compute a new ω and a new $S_E(\mathcal{S}_\omega)$. We then compute ρ_{n+1}, γ_{n+1}, and $u^{(n+1)}$ and are ready for the next iteration.

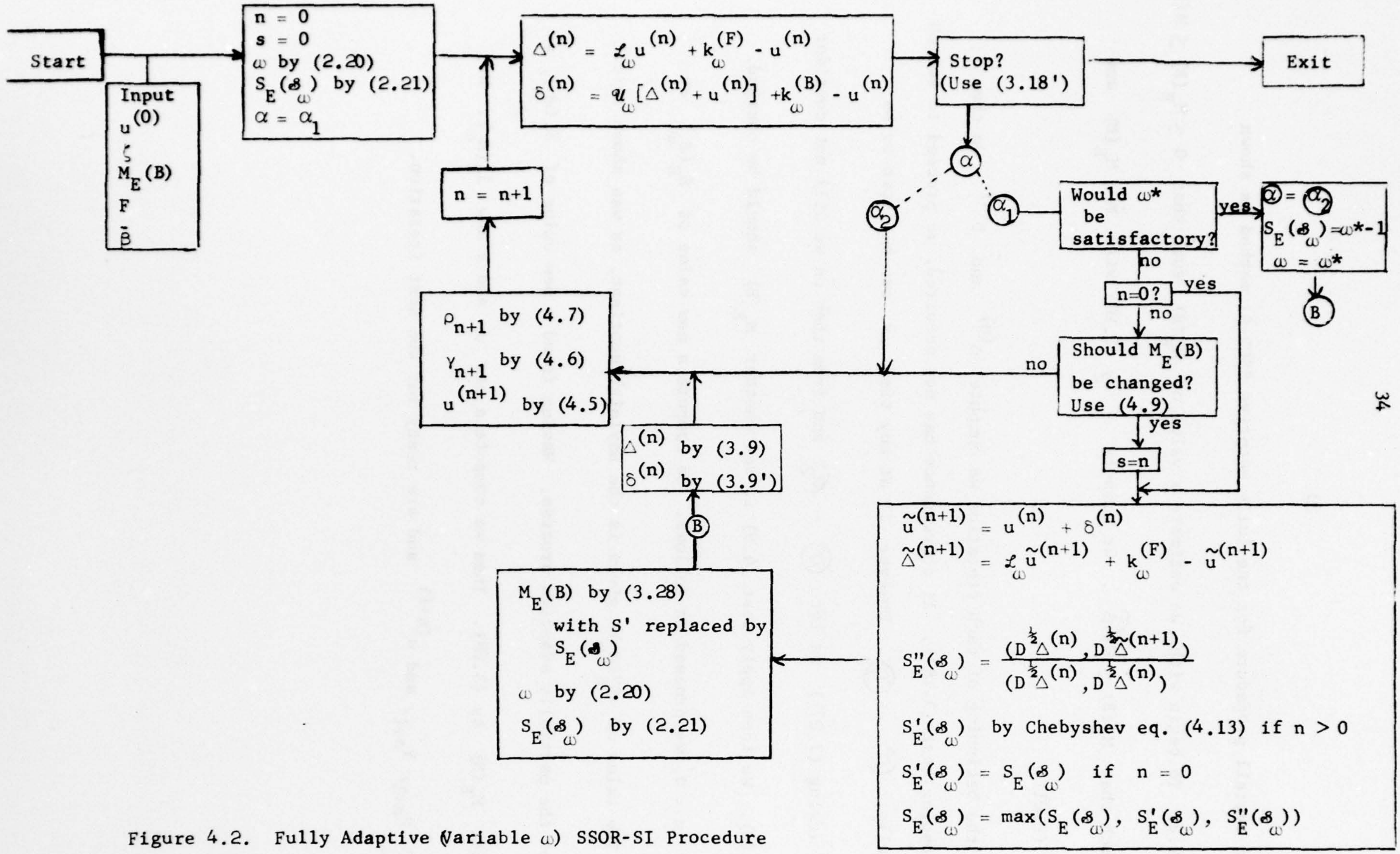

Figure 4.2. Fully Adaptive (Variable ω) SSOR-SI Procedure

5. <u>Numerical results</u>. Several experiments were run for the SSOR method using the conjugate-gradient adaptive scheme and using the semi-iterative fully adaptive scheme. The following problem, which we call "model problem P," was used as a test case.

$$(5.1)\qquad \begin{cases} \Delta u = -1 & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \\ \Omega = (0,1)\times(0,1). \end{cases}$$

Numerical experiments on more general problems, such as those considered in Young [1974] are now being carried out.

The five-point finite difference method was used to generate the matrix problem

$$(5.2)\qquad Au = b$$

for mesh sizes of $h = 1/20$, $1/40$, and $1/80$. For these cases the parameters $\bar{\mu}$, ω_1, and the bound for $S(\mathcal{S}_{\omega_1})$, as given by (2.30), (2.31), and (2.32) (with $\bar{\beta} = 1/4$), respectively, are shown in Table 1.

TABLE 1
VALUES OF OPTIMUM PARAMETERS FOR MODEL PROBLEM P

	$h = 1/20$	$h = 1/40$	$h = 1/80$
$\bar{\mu}$	.98769	.99692	.99923
ω_1	1.72874	1.85445	1.92448
$S(\mathcal{S}_{\omega_1})$	.85451	.52448	.96151

In our test cases, since we were interested in studying the effectiveness of the adaptive schemes rather than studying effectiveness of stopping procedures, we generated the exact solution, $\bar{u}$, to the problem (5.2) and iterated until the following condition was satisfied:

(5.3) $$K_{exact} = \frac{\|u^{(n)} - \bar{u}\|}{\|\bar{u}\|} \le \zeta = 10^{-6}.$$

The adaptive procedures, as described in Sections 3 and 4, were used for the SSOR-CG and for the SSOR-SI schemes, respectively. The stopping test (5.3) rather than (2.18') was used in both cases. For the SSOR-SI method only the fully adaptive scheme was used. No test cases were run using the partially adaptive procedure. For the adaptive process, an initial guess $\bar{\mu}_E^{(0)}$ was made for $\bar{\mu} = S(B)$. It is necessary that $\bar{\mu}_E^{(0)} \le \bar{\mu}$, and in the absence of a better guess $\bar{\mu}_E^{(0)}$ can be set equal to zero. Initial estimates for ω and $S(\mathcal{S}_\omega)$ were obtained from (2.20) and (2.21) using the estimated parameter $\bar{\mu}_E^{(0)}$. The adaptive procedures of Sections 3 and 4 were then used to correct the estimated parameters.

A non-adaptive process was also used for both methods. In the non-adaptive process a value of ω was chosen and not changed. In the case of the non-adaptive SSOR-SI method a value of $S(\mathcal{S}_\omega)$ was used based on (2.21) with $\bar{\mu}_E$ replaced by the true value of $\bar{\mu}$ and with $\bar{\beta} = 1/4$.*

Table 2 gives the results for the adaptive and non-adaptive procedures where $\omega = .82$ was used for the non-adaptive process. (This corresponds to using (2.20) with $\bar{\mu}_E = 0$ and $\bar{\beta} = 1/4$.) For the adaptive cases $\bar{\mu}_E$ was initially zero. The value $F = 3/4$ was used for the adaptive cases.

For the values given for the adaptive SSOR-CG method, the numbers in parentheses correspond to a different adaptive scheme where at least two iterations were used with each parameter set. The numbers look somewhat better than in the original adaptive case.

---

*Clearly, this procedure is more favorable to the non-adaptive SSOR-SI method than would be the case if we had used the more reasonable procedure of choosing a value of $\bar{\mu}_E$ and then calculating ω and a bound for $S(\mathcal{S}_\omega)$ by (2.18), letting $\bar{\beta} = 1/4$.

TABLE 2. COMPARISON OF ADAPTIVE AND NON-ADAPTIVE SCHEMES

	SSOR-SI		
	h = 1/20	h = 1/40	h = 1/80
Non-adaptive			
$\omega = .82$	32	67	120
optimum parameters*	17	25	35
Fully adaptive			
$\bar{\mu}_E^{(0)} = 0$	23	26	39

	SSOR-CG		
	h = 1/20	h = 1/40	h = 1/80
Non-adaptive			
$\omega = .82$	17	28	52
optimum parameters*	12	17	23
Adaptive			
$\bar{\mu}_E^{(0)} = 0$	16 (14)	21 (20)	32 (27)

*See Table 1

Figure 5.1 shows the number of iterations required to satisfy the convergence criteria (5.3) for the non-adaptive SSOR-CG method and for the non-adaptive SSOR-SI method. In each case a value of ω was chosen and the non-adaptive procedure was used as described above. It should be noted that the SSOR-CG curve is much flatter than the SSOR-SI curve, and also lies considerably below the SSOR-SI curve. This indicates that the SSOR-CG procedure is much less sensitive to choice of ω. The advantage of the SSOR-CG method would probably have been even greater if we had not used the true value of $\bar{\mu}$ in computing the bound on $S(\mathcal{S}_\omega)$ for the non-adaptive SSOR-SI scheme.

The results presented in Tables 3 and 4 show how the parameters $\bar{\mu}_E$, ω, and $S(\mathcal{S}_\omega)_E$ changed during the adaptive process. In the cases considered, the initial $\bar{\mu}_E^{(0)}$ was set equal to zero. It is evident that after only two or three iterations, good values of the parameters are available through the adaptive process. Furthermore, even the iterations occurring before the selection of the final parameter set are not "wasted" iterations since the iterant $u^{(n)}$ is being improved, although not as much as would be the case with the optimum parameters.

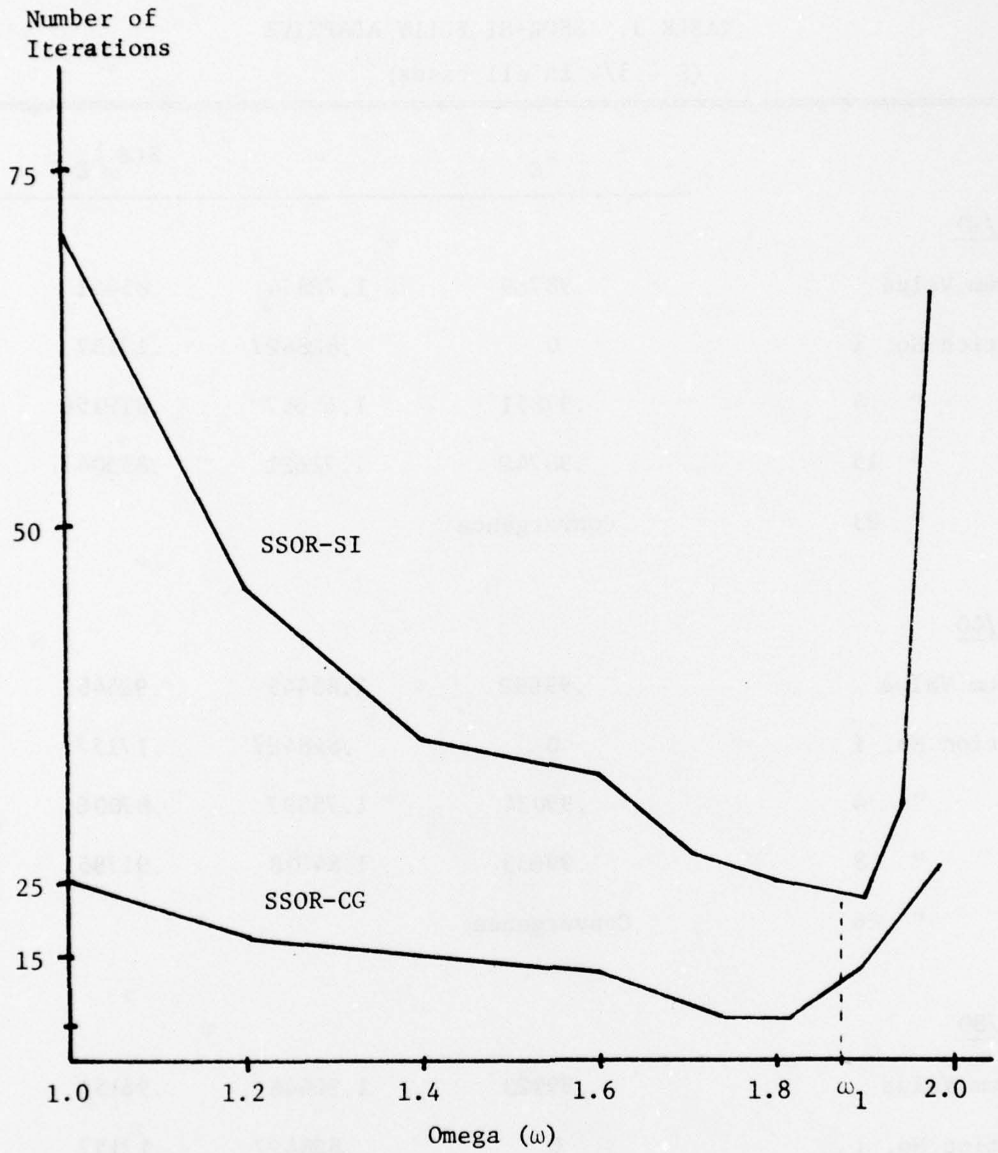

Figure 5.1. Model Problem P. # Iterations vs. ω, $h = 1/40$

Non-adaptive Schemes

TABLE 3. SSOR-SI FULLY ADAPTIVE

(F = 3/4 in all cases)

	$\bar{\mu}_E$	ω	$S(\mathcal{L}_\omega)_E$
h = 1/20			
Optimum Value	.98769	1.72874	.85451
Iteration No. 1	0	.828427	.17157
" " 4	.97851	1.65657	.812156
" " 15	.98742	1.72621	.853047
" " 23	Convergence		
h = 1/40			
Optimum Value	.99692	1.85445	.92448
Iteration No. 1	0	.828427	.17157
" " 4	.99034	1.75597	.87006
" " 8	.99633	1.84218	.91785
" " 26	Convergence		
h = 1/80			
Optimum Value	.99923	1.92448	.96151
Iteration No. 1	0	.828427	.17157
" " 4	.99540	1.82504	.90852
" " 8	.99896	1.91280	.95543
" " 39	Convergence		

TABLE 4. SSOR-CG ADAPTIVE

(F = 3/4 in all cases)

	$\bar{\mu}_E$	ω	$S(\mathcal{L}_{\omega_E})$
h = 1/20, k = 1			
True Value	.98769	1.72874	.85451
Iteration No. 1	0	.828427	.17157
" " 3	.97851	1.65657	.812156
" " 7	.98742	1.72621	.853047
" " 16	Convergence		
h = 1/40, k = 2			
True Value	.99692	1.85445	.92448
Iteration No. 1	0	.82847	.17157
" " 4	.99137	1.77713	.871171
" " 20	Convergence		
h = 1/80, k = 1			
True Value	.99923	1.92448	.96151
Iteration No. 1	0	.82847	.17157
" " 3	.99348	1.79784	.89047
" " 7	.99881	1.91429	.94801
" " 32	Convergence		

Many cases were run for various values of $\bar{\mu}_E^{(0)}$ and for various values of F. Figures 5.2 and 5.3 illustrate the effect of varying $\bar{\mu}_E$ in the adaptive procedure. The solid lines represent the non-adaptive case; i.e., $\bar{\mu}_E^{(0)}$ is chosen and the parameters are calculated based on $\bar{\mu}_E^{(0)}$ and are not changed (ω is fixed throughout the procedure). The dashed line represents the fully adaptive procedure; here $\bar{\mu}_E^{(0)}$ is specified and all parameters are updated when the improved parameter set is calculated. The ordinate represents the number of iterations required for convergence and the abscissa is $\bar{\mu}_E^{(0)}$. The graphs are drawn for $h = 1/40$, but these results are representative of all of the test cases

In general, lowering the value of F means there will be fewer parameter changes, and setting F closer to unity has the effect of fine-tuning the procedure. In this case there will be more parameter changes and the final set should be closer to the optimum values. However, when the parameters are changed, the convergence rate of the overall method drops off initially and then gradually increases. Thus, changing parameters too often can have the net effect of lowering the overall convergence rate. In choosing F, one must balance the effect of changing parameters against the gain which will be realized from improved parameters. Figure 5.4 is a graph of the number of iterations required versus the parameter F for the SSOR-CG adaptive procedure and for the SSOR-SI fully adaptive procedure. In general, this procedure does not seem to be too sensitive to values of F as long as they are not close to zero or to one.

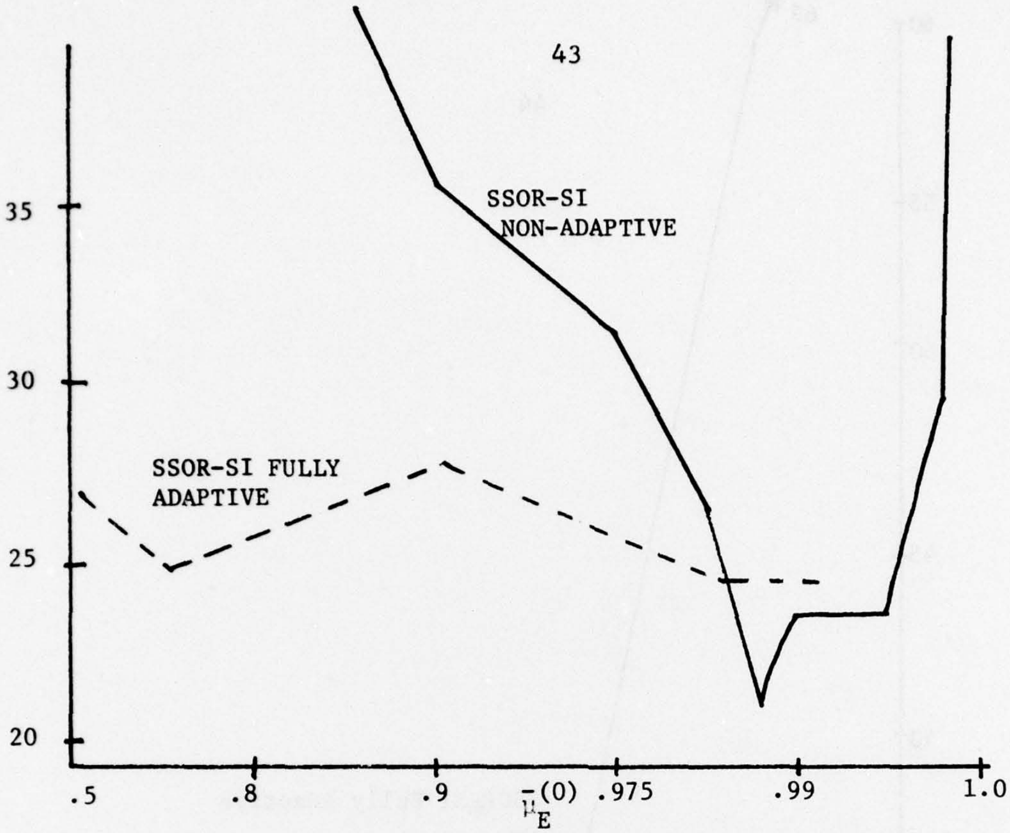

Figure 5.2. Iterations vs. $\bar{\mu}_E^{(0)}$ Fully Adaptive and Non-adaptive SSOR-SI, h = 1/40

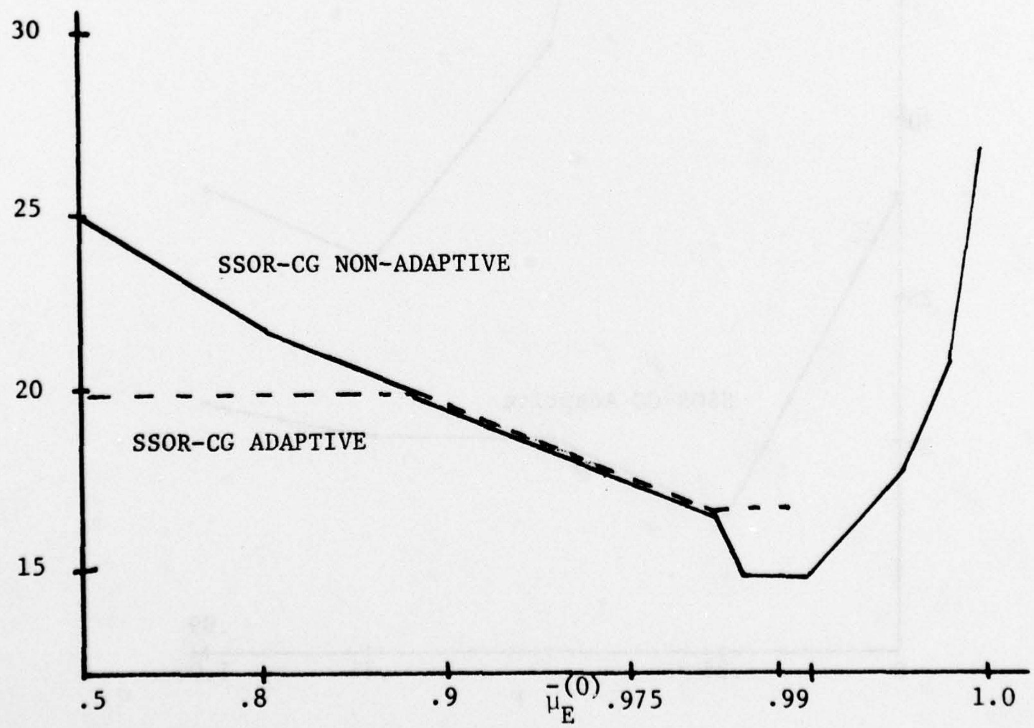

Figure 5.3. Iterations vs. $\bar{\mu}_E^{(0)}$ Adaptive and Non-adaptive SSOR-CG, h = 1/40

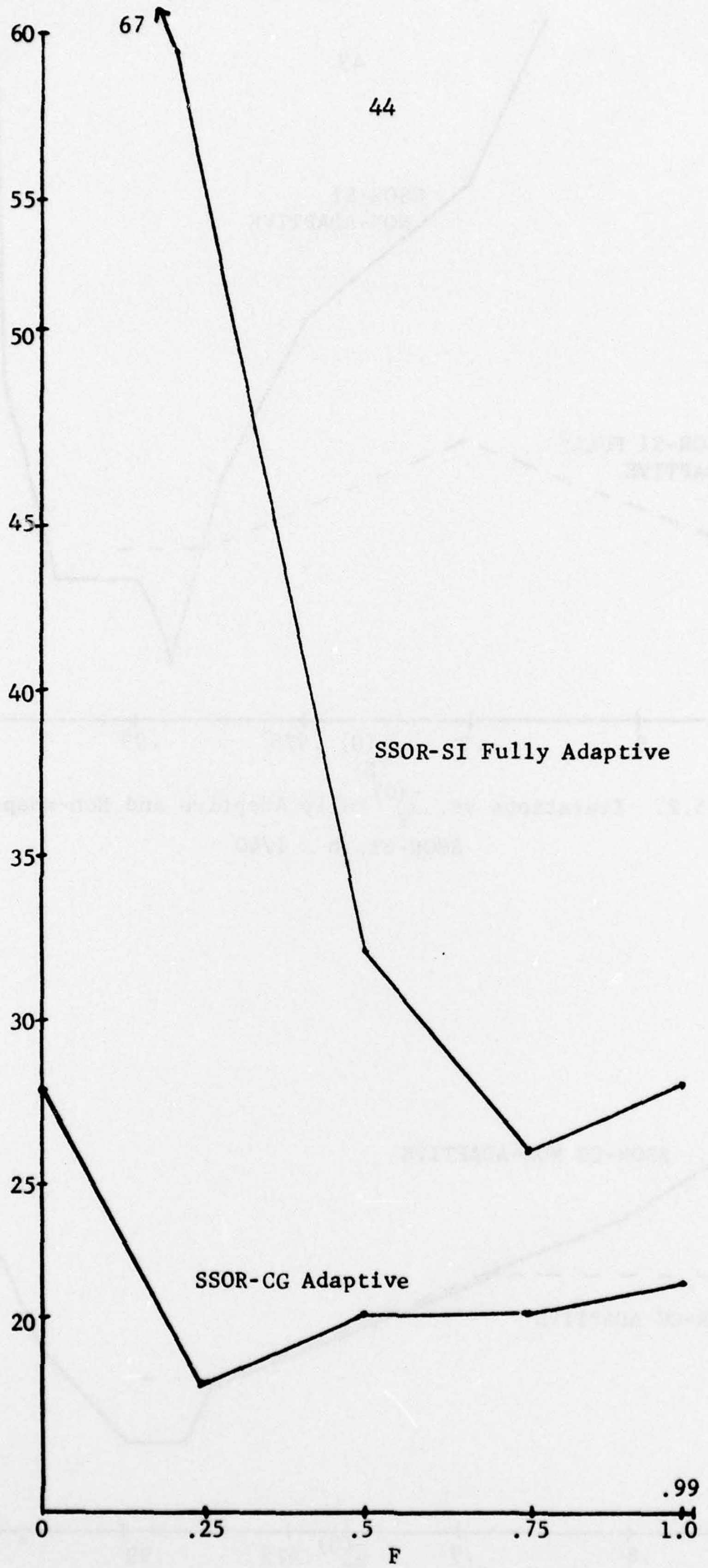

Figure 5.4. Number of Iterations vs F for SSOR-CG and SSOR-SI Adaptive ($h = 1/40$, $\mu_E^{(0)} = 0.$, $\bar{\beta} = 1/4$)

6. Summary and conclusions. These preliminary test cases seem to indicate that the adaptive procedures presented in this paper are effective when they are coupled with both the SSOR-CG and the SSOR-SI methods. It has been shown that for this problem very few iterations are required to obtain a good set of parameters, and these preliminary iterations are not wasted, in the sense that the vector $u^{(n)}$ is being improved on each iteration. The procedures are effective even when the initial guess of $\bar{\mu}_E$ is the worst possible; i.e., $\bar{\mu}_E^{(0)} = 0$. It has been shown that in this case the number of iterations using the adaptive scheme is not too many more than the number of iterations which would have been required if the optimum parameters had been known from the start.

These results are preliminary in the sense that numerical experiments have been carried out only for the model problem P. We are now testing these procedures on a more general class of self-adjoint elliptic problems.

References

1. David M. Young and R. T. Gregory, A Survey of Numerical Mathematics, Vol. II, Addison-Wesley, Reading, Mass., 1973.

2. David M. Young, "On the accelerated SSOR method for solving large linear systems," Advances in Mathematics 23 (1977), 215-271. (Also CNA-92, Center for Numerical Analysis, UT Austin, September 1974.)

3. David M. Young, Iterative Solution of Large Linear Systems, Academic Press, New York, 1971.

4. L. A. Hageman and David M. Young, "On the effective use of iterative methods for solving large linear systems," in preparation.

5. V. J. Benokraitis, "On the Adaptive Acceleration of Symmetric Successive Over-relaxation," doctoral thesis, UT Austin, 1974.

6. P. Concus, G. H. Golub, and D. P. O'Leary, "A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations," in Sparse Matrix Computations, J. R. Bunch and D. J. Rose, eds., Academic Press, 1976.

7. D. P. O'Leary, "Hybrid Conjugate Gradient Algorithms," doctoral thesis, Computer Sciences Department, Stanford University, 1975.

8. S. Kaniel, "Estimates for some computational techniques in linear algebra," Math. Comp. 20 (1966), 369-378.

9. C. C. Paige, "The Computation of Eigenvalues and Eigenvectors of Very Large Sparse Matrices," doctoral thesis, London University, Institute of Computer Science, 1971.

10. O. Axelsson, "On preconditioning and convergence acceleration in sparse matrix problems," Report CERN 74-10 of the CERN European Organization for Nuclear Research, Data Handling Division, Laboratory I, 8 May 1974.

11. David M. Young and Linda J. Hayes, "Notes on the conjugate gradient method," CNA report in preparation.

12. David M. Young, Linda J. Hayes, and Edward E. Schleicher, "The use of the Accelerated SSOR method to solve large linear systems," Abstract, Fall SIAM Meeting, December 1975, San Francisco, California.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
~~ITR CNA-123~~ 12301.6-M

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
The Accelerated SSOR Method for Solving Large Linear Systems

5. TYPE OF REPORT & PERIOD COVERED

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Linda J. Hayes and David M. Young

8. CONTRACT OR GRANT NUMBER(s)
DAHC04 74 G 0198

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Center for Numerical Analysis
The University of Texas at Austin
Austin, Texas 78712

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
P-12301-M

11. CONTROLLING OFFICE NAME AND ADDRESS
The University of Texas at Austin
Austin, Texas 78712

12. REPORT DATE
May 1977

13. NUMBER OF PAGES
46

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)
unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES
The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
iterative, symmetric successive overrelaxation, Chebyshev acceleration, conjugate gradient , adaptive parameter determination

omega

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The symmetric SOR method (SSOR-method) for solving the linear system Au = b is considered. The basic properties of the SSOR method are summarized, and a procedure is given for estimating the optimum relaxation factor ω and the corresponding spectral radius of the SSOR matrix S_ω. Two procedures for accelerating the convergence of the SSOR method are considered, one based on conjugate gradient acceleration and the second based on the use of Chebyshev acceleration. Two versions of conjugate gradient acceleration are considered--the nonadaptive and the adaptive.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

S sub omega